technische universität dortmund

SoSe 2017

C. Sohler
A. Krivošija, A. Rey, H. Sandvoß

DAP2 – Heimübung 9

Ausgabedatum: 9. 6. 17 — Abgabedatum: Montag 19. 6. für alle Gruppen, 12 Uhr

Schreiben Sie unbedingt immer Ihren **vollständigen Namen**, **Ihre Matrikelnummer** und **Ihre Gruppennummer** auf Ihre Abgaben!

Aufgabe 9.1 (5 Punkte): (Dynamische Programmierung)

Es seien $a = (a_1, \dots, a_n) \in \{0,1\}^n$ und $b = (b_1, \dots, b_m) \in \{0,1\}^m$ zwei Bitstrings der Längen n und m. Um den Abstand der beiden Bitstrings zu bestimmen, werden Einfüge-, Lösch- und Ersetzungsoperationen betrachtet, die einen String in den anderen umwandeln. Diese Operationen haben nun unterschiedlichen Kostenaufwand: Eine 0 hinzuzufügen oder zu entfernen ist kostenlos, eine 1 durch eine 0 zu ersetzen oder umgekehrt kostet 2 Einheiten. Beispielsweise kostet es also 2 Einheiten, $a = (1,1,1,0)$ in $b = (1,0,1)$ umzuwandeln, indem man a_2 durch 0 ersetzt (Kosten 2) und $a_4 = 0$ löscht (Kosten 0).

a) Welches ist der minimale Kostenaufwand, um $c = (0,1,1,0)$ in $b = (1,0,1)$ umzuwandeln?

b) Geben Sie eine rekursive Form $K(i,j)$ zur Berechnung des minimalen Kostenaufwands an, um einen Bitstring $a = (a_1, \dots, a_i)$ in einen Bitstring $b = (b_1, \dots, b_j)$ umzuwandeln.

c) Zeigen Sie die Korrektheit der in Teilaufgabe b) angegebenen rekursiven Form für einen beliebigen String $a = (a_1, \dots, a_i)$ und den Fall, dass $b = (b_1, \dots, b_j)$ mit $b_k = 0$ für alle k, $1 \le k \le j$, ist.

Aufgabe 9.2 (5 Punkte): (Binäre Bäume und Suchbäume)

Neben dem kennengelernten *Inorder*-Durchlauf, betrachtet ein *Preorder*-Durchlauf zuerst die Wurzel und danach den linken und rechten Teilbaum jedes Teilbaums. Umgekehrt betrachtet ein *Postorder*-Durchlauf zuerst den linken, dann den rechten Teilbaum und danach die Wurzel jedes Teilbaums.

Wir nennen einen Binärbaum der Höhe h *vollständig*, falls alle Knoten zwei oder keine Kinder besitzen und alle Blätter den gleichen Abstand h zur Wurzel haben. Ein Knoten w heißt (direkter) Nachfolger eines Knotens v, falls der Schlüssel $S(w)$ von w größer als der Schlüssel $S(v)$ von v ist und kein Knoten x existiert mit Schlüssel $S(x)$, $S(w) > S(x) > S(v)$. Ein Knoten heißt *innerer Knoten*, falls er kein Blatt is.

a) Geben Sie einen binären Baum an, sodass ein *Inorder*-Durchlauf die gleiche Knotenfolge wie ein *Postorder*-Durchlauf liefert. Geben Sie außerdem einen binären Baum an, sodass ein *Inorder*-Durchlauf die gleiche Knotenfolge wie ein *Preorder*-Durchlauf liefert.

b) Zeigen Sie mit struktureller vollständiger Induktion, dass ein vollständiger binärer Baum 2^h Blätter besitzt.

c) Zeigen Sie, dass in einem vollständigen binären Suchbaum der direkte Nachfolger eines inneren Knotens immer ein Blatt ist.